

**Method and System for Time Synchronization****RECEIVED**

DEC 12 2001

Technology Center 2600

Field of the Invention

[001] This invention relates to time synchronization, and in particular to a method for synchronizing a time clock of a client station via a communications network having an unknown transmission time.

RECEIVED

JAN 08 2002

Background of the Invention

Group 2600

[002] Most conventional computer systems maintain an internal clock to keep track of the time of day. Accurate time of day measurements are required in a wide variety of applications such as managing and tracking electronic mail, timing back-ups of data on a network, synchronizing communications between clients and servers, managing multimedia teleconferences, and assigning a correct time to documents or transactions. Since the internal clock is not perfectly accurate, as time goes by the internal time maintained by the internal clock diverges from the external time with which the internal clock was previously synchronized. This time divergence is sometimes called "clock drift". Typically the internal clock's time drifts away from the time as a linear function of elapsed time since synchronization. To prevent the clock drift from getting too large, from time to time the internal clock is resynchronized with the external reference time. Time services exist which provide accurate time information using an "atomic clock". One well-known time service is WWV, which broadcasts a Universal Time Signal. Various solutions have been developed to synchronize the time clock of a computer system. A simple method is for a user of the computer system to manually adjust the clock whenever the clock appears to have drifted. This technique, however, is both inconvenient for the user and subject to its own inaccuracies.

[003] Also, time sources such as WWV occasionally introduce "leap seconds" to synchronize their time with the motions of the planet Earth. To prevent error from accumulating due to drift and leap seconds, it is particularly desirable to synchronize the internal clock of a

RECEIVED

DEC 20 2001

Technology Center 2600

computer system with the external reference time automatically without intervention of a user or network administrator.

[004] Some prior art methods for synchronizing the internal clock employ a round trip scheme in which a computer system sends a synchronization request to a time service and the time service responds by sending a synchronization message. The inaccuracy of time provided to a computer system is directly related to the total elapsed time for the round trip sequence of messages. Thus, the precision with which clock synchronization can be achieved is limited by the time required for the round trip sequence.

[005] A particular type of round trip synchronization, called Probabilistic Clock Synchronization has been used for synchronizing internal times of computer systems with a reference time from an external source via a communications network. The technique is described in Cristian, " Probabilistic Clock Synchronization", IBM Technical Disclosure Bulletin, Vol. 31, No. 2 (July 1988), p. 91 which is incorporated herein by reference. The basic round trip sequence works as follows: A client station sends a synchronization request at a time t , according to its clock. A time server responds with a message giving a time T , according to the server's time clock. The client station receives the response at a time t' . It is thus established that the server's time T falls somewhere within the time interval between the client stations times t and t' . Preferably, T is synchronized with the midpoint of the interval between t and t' . Thus, the precision of the client station's synchronization is accurate to within $(t' - t)/2$. If the achieved precision is not considered good enough, the round trip message exchange sequence is repeated. This method provides means for taking the roundtrip delay time into account and, therefore, reduce the error occurring due to the transmission. However, this method requires a substantial amount of processing in real time.

[006] An improved version of "Cristian's" method is disclosed by Cheung et al. in U.S. Patent No. 5,535,217 issued July 9, 1996, which is incorporated herein by reference. Enhanced precision is achieved by computing a new precision range for the synchronized time based on an intersection between precision intervals of the client station's time and the time server's time. Unfortunately, also this method requires a substantial amount of processing in real time.

Furthermore, both methods determine the time elapsed to transmit a message from the time server to the client station as half the round trip delay.

[007] In U.S. Patent No. 6,023,769 issued February 8, 2000, Gonzalez discloses a method and apparatus for synchronizing an imprecise time clock maintained by a computer system wherein a first time reading is obtained from a local timing service at a precision greater than that of the clock, and a second time reading is obtained from a remote time server at a precision greater than that of the clock. The second time reading from the time server is corrected for delays associated with the transmission over the network using the first time reading from the local timing service. This method may be able to give a more correct estimate of the time elapsed to transmit a message from the remote time server to the client station. However, it requires messages being transmitted between the client station and the local timing service at a known transmission time or at a time less than the precision error of the local timing service.

[008] Furthermore, none of these methods provide means for preventing tampering with the messages, for example, deliberately delaying the messages by a third unauthorized party.

[009] It is an object of the invention to provide a method for synchronizing a timing device of a client station that substantially reduces the error due to round trip delay.

[0010] It is further an object of the invention to provide a method for synchronizing a timing device of a client station that substantially reduces real time processing at the time server as well as at the client station.

[0011] It is yet another object of the invention to provide a method for synchronizing a timing device of a client station that substantially reduces the risk of tampering by an unauthorized third party.

Summary of the Invention

[0012] In accordance with the invention there is provided, a method for synchronizing a timing device of a client station. The method comprises the steps of:

sending a plurality of packets, each packet being sent at a predetermined time, from a time server to the client station via a communications network;

receiving the plurality of packets at the client station;

determining a time indicative of a local time of receipt of the plurality of packets at the client station and storing time data in dependence thereon;

returning the plurality of packets to the time server via the communications network;

determining a time indicative of a local time of receipt of the plurality of packets at the time server; and,

determining data in dependence upon round trip delay of the packets and variance in packet spacing and comparing the data to threshold values.

[0013] In accordance with the invention there is provided, a method for synchronizing a timing device coupled to a communications network. The method comprises the steps of:

sending a plurality of packets, each packet being sent at a predetermined time, from a first node to a second node via a communications network;

receiving the plurality of packets at the second node;

determining a time indicative of a local time of receipt of the plurality of packets and providing time data in dependence thereon;

returning the plurality of packets to the first node via the communications network;

determining a time indicative of a local time of receipt of the plurality of packets at the first node; and,

determining data in dependence upon round trip delay of the packets and variance in packet spacing and comparing the data to threshold values.

[0014] In accordance with the invention there is further provided, a system for synchronizing a timing device of a client station via a communications network. The system comprises:

first processing means connected to the communications network and a time clock;
and, second processing means connected to the communications network and the timing device,

wherein the first processing means comprise:

means for sending in real time a plurality of packets, each packet being sent at a predetermined time, via the communications network to second processing means;

means for receiving the plurality of packets from the second processing means in real time;

means for determining in real time a time indicative of a time of receipt of each of the plurality of packets;

means for determining data in dependence upon round trip delay of the packets and variance in packet spacing and comparing the data to threshold values;

means for determining data indicative of a correct time in dependence upon time data from the time clock and the data in dependence upon round trip delay of the packets and variance in packet spacing; and,

means for sending a signal comprising the data indicative of a time correction to the second processing means via the communications network;

and,

wherein the second processing means comprise:

means for receiving a plurality of packets from the first processing means in real time;

means for determining in real time a time indicative of a time of receipt of each of the plurality of packets;

means for returning the plurality of packets in real time to the first processing means;

means for receiving a signal comprising the data indicative of a time correction from the first processing means; and,

means for synchronizing the timing device in dependence upon the data indicative of the time correction.

Brief Description of the Drawings

[0015] Exemplary embodiments of the invention will now be described in conjunction with the drawings, in which:

[0016] Fig. 1a is a schematic diagram of a client station connected to a time server via a communications network according to the invention;

[0017] Fig. 1b is a simplified flow diagram of a method for synchronizing a timing device of a client station according to the invention;

[0018] Fig. 1c is a timing diagram illustrating timing of the plurality of pulses according to the disclosed method for synchronizing a timing device;

[0019] Fig. 2a is a schematic diagram of a plurality of client stations connected to a time server via a communications network;

[0020] Fig. 2b is a simplified flow diagram of another embodiment of the method for synchronizing a timing device of a client station according to the invention;

[0021] Fig. 3 is a simplified flow diagram of another embodiment of the method for synchronizing a timing device of a client station according to the invention;

[0022] Fig. 4 is a simplified flow diagram of another embodiment of the method for synchronizing a timing device of a client station according to the invention;

[0023] Fig. 5 is a simplified flow diagram of yet another embodiment of the method for synchronizing a timing device of a client station according to the invention;

[0024] Fig. 6 is a simplified flow diagram of another method for synchronizing a timing device of a client station according to the invention;

[0025] Fig. 7a is a simplified flow diagram of a method for determining data indicative of a time correction according to the invention;

[0026] Fig. 7b is a simplified flow diagram of another method for determining data indicative of a time correction according to the invention;

[0027] Fig. 7c is a simplified flow diagram of another method for determining data indicative of a time correction according to the invention;

[0028] Fig. 7d is a timing diagram illustrating evaluation of the timing of the plurality of pulses according to the method shown in Fig. 7c;

[0029] Fig. 7e is a simplified flow diagram of another method for determining data indicative of a time correction according to the invention;

[0030] Fig. 7f is a timing diagram illustrating evaluation of the timing of the plurality of pulses according to the method shown in Fig. 7e; and,

[0031] Fig. 7g is a simplified flow diagram of yet another method for determining data indicative of a time correction according to the invention;

Detailed Description of the Invention

[0032] Referring to Figs. 1a and 1b, a method for synchronizing a timing device 4 of a client station 1 according to the invention is shown. The client station 1 of system 10 is connected to a time server 3 via a communications network 2 such as the Internet. The time server 3 sends a plurality of digital signals (hereinafter, referred to simply as "packets"), each comprising m bits, at predetermined times via the communications network 2 to the client station 1, shown in Fig. 1b as step 101. Preferably, each packet comprises an identifier, for example the number of the packet, and an identifier of the time server 3 in the form of at least a random number. Upon receipt 102 of each of the packets at the client station 1, a time indicative of a local time of receipt of each of the packets is determined and time data in dependence thereon are stored 104. The packets are then returned 106 to the time server 3 via the communications network 2, where upon receipt 107 a time indicative of a local time of receipt of each of the plurality of packets is determined 109. Fig. 1c shows a distribution of the packets versus a time axis for steps 102, 106, and 107, respectively, indicated by I, II, and III. At a predetermined time t the server sends n packets 5 such that time intervals Δt between consecutive packets are a same, shown in diagram I. As is obvious to a person of skill in the art, same time intervals between consecutive packets is preferred but not required for the invention. Diagram II shows the distribution of the n packets 5 upon receipt at the client station 1 versus a local time axis t' . Due to random variations of the transmission

speed of each of the packets, the time intervals between consecutive packets will be different, as indicated by $\Delta t'$, $\Delta t''$, $\Delta t'''$, In diagram III the distribution of the n packets 5 upon receipt at the time server 3 versus a local time axis t is illustrated. The first packet arrives at a time t^* , which is the predetermined time t plus a round trip delay for the first packet. Time differences between consecutive packets arriving at the time server are again different and differ from the respective time differences in diagram II as well, as indicated by Δt^* , Δt^{**} , Δt^{***} , Variations in the round trip delays of the n packets provides useful information about network behavior during the transmission such as random variations of the transmission speeds or a slow down of the communications network 2, if the number n of the packets is sufficiently large (for example: $n = 100$).

[0033] In step 111 the packets are analyzed in order to determine data in dependence upon round trip delay and variations in spacing between consecutive packets. These data are then compared to threshold values 113 and if the data are within the threshold values the process of time correction is continued. In step 117 data indicative of a time correction are determined and a signal comprising the data indicative of a time correction is sent from the time sever 3 to the client station 1 as shown in step 119. Methods for determining the data of a time correction according to the invention will be described in detail at a later point. Upon receipt 120 of the signal comprising the data indicative of a time correction the timing device 4 of the client station 1 is synchronized accordingly 122.

[0034] If the data in step 113 are not within the threshold values, the steps 101 to 113 are repeated or an error indicator is sent 115 notifying a user of the client station 1 or a network administrator. Preferably, the threshold values are determined using a statistical evaluation of the round trip delays within the network. This allows detection of any irregularities in the communications network such as technical problems or an attack of a "Man in the Middle" delaying transmission of the packets.

[0035] Steps 101 to 109 of the method according to the invention are performed in real time. These steps do not require much processing and can easily be performed by a processor in

parallel to other tasks. The more elaborate steps 111 to 117 may be performed at any time, for example, when workload of a processor is reduced. Therefore, the method according to the invention obviates the need for large processing capacity of the time server 3 compared to the prior art.

[0036] In another embodiment of the method according to the invention the time server 3 sends 201 a plurality of packets to all nodes of a communications network 2, as shown in Figs. 2a and 2b. Non responding nodes are indicated by circles in Fig. 2a. Responding client stations 1A to 1D, for example, attach 202 to at least one of the packets an authenticator identifying the client station before returning them to the time server 3. The time server 3 extracts the authenticator from each plurality of packets received from the client stations 1A to 1D in step 203 in order to identify each responding client station before further processing the packets accordingly. A signal comprising the data indicative of a time correction is then sent 219 to each identified client station. This method further reduces the workload of the processor of the time server 3 by sending only one plurality of packets for synchronizing a plurality of clients 1A to 1D.

[0037] Referring to Fig. 3, another embodiment of the method according to the invention is shown. Each of the plurality of packets sent 301 from the time server comprises a tamper proof identifier of the time server 3. For example, a tamper proof identifier includes a random number or a signed identification as is obvious to a person of skill in the art. The client station compares the identifier with a reference 302 and if the time server 3 is identified returns the packets 106. If the time server cannot be identified the synchronization process is stopped by the client station. Optionally, the client station 1 sends a warning signal 306 to the time server 3 or a network administrator when server identification fails. The time server 3 itself compares the identification with a reference in step 307 and continues with the synchronization process if the packets have been identified 309 as the ones sent by the time server 3. If the packets cannot be identified 309 by the time server 3 the synchronization process is optionally repeated with step 301 and/or the time server may sent a warning message 311. Therefore, this method substantially enhances security by providing means to reliably recognize if the packets had been tampered with by an unauthorized third party.

[0038] Security is further enhanced by securely signing the data indicative of a time correction, as shown in step 401 in Fig. 4. The client station 3 compares the signature with a reference 402 and synchronizes its timing device 4 only after the comparison is indicative of a match 404. Methods for securely signing the data, for example, encryption algorithms are well known in the art.

[0039] Fig. 5 shows yet another embodiment of the method according to the invention. In step 502 each packet is time stamped at the client station 1 with data indicative of a local time of receipt of the packet at the client station 1. This method provides valuable information to the time server 3 which is there extracted 503 for a more accurate estimate of a transmission time needed by a packet to travel from the time server 3 to the client 1, thus enhancing accuracy of the synchronization. The method for determining the transmission time according to the invention is discussed hereinbelow.

[0040] Fig. 6 illustrates another method for synchronizing a timing device 4 of a client station 1 according to the invention. The client station 1 sends 601 a plurality of packets, each at a predetermined time via communications network 2 to time server 3. Upon receipt 602 of the packets the time server 3 determines 604 data indicative of a correct time of receipt of each of the plurality of packets and time stamps 606 each of the packets with the data indicative of the correct time of receipt of the packet. The time stamped packets are then returned 608 to the client station 1 via the communications network 2. Upon receipt 609 of the packets the client station 1 determines 611 data indicative of a local time of receipt for each of the plurality of packets. In step 613 the packets are analyzed in order to determine data in dependence upon round trip delay and variations in spacing between consecutive packets. These data are then compared to threshold values 615 and if the data are within the threshold values data indicative of a time correction are determined 617. In step 619 the timing device 4 of the client station 1 is synchronized according to the data indicative of a time correction.

[0041] This method requires only minimal processing capabilities for the time server 3, all steps concerning data evaluation are done by the client station. Furthermore, only steps 601 to

611 have to be performed in real time, leaving the elaborate steps of data evaluation - steps 613 to 619 - for times of minimum workload of the client station 1.

[0042] As is evident to a person of skill in the art, all embodiments discussed with reference to Figs. 3, 4, and 5 are also applicable to this method according to the invention as well as the following embodiments for the determination of data indicative of a time correction.

[0043] In the following, various methods according to the invention for determining data indicative of a time correction used in step 117 are described.

[0044] A first simple method for determining data indicative of a time correction is illustrated in Fig. 7a. A round trip delay value for each of the plurality of packets is determined as difference between the time each packet has been received at the time server 3 and the predetermined time each packet has been sent from the time server 3. A minimum round trip delay value is determined among the round trip delay values of each of the plurality of packets. Data indicative of a correct time are then calculated by adding half of the minimum round trip delay value to the predetermined time the packet having the minimum round trip delay value has been sent by the server.

[0045] Fig. 7b shows an improved embodiment of the previous method, which is preferable if the round trip delays of the packets vary considerably. Here, interpolation using at least two round trip delay values is applied in order to determine a more accurate minimum round trip delay value. In an additional step, a predetermined time corresponding to the interpolated minimum round trip delay value is calculated. Data indicative of a correct time are then calculated by adding half of the interpolated minimum round trip delay value to the interpolated predetermined time.

[0046] The assumption that the transmission time of a packet from the time server 3 to the client 1 station is half the round trip time of the packet does not take into account variations of the transmission speed between the transmission to the client station 1 and the transmission to the time server 3. Therefore, use of these methods can result in rather inaccurate time

corrections. More accurate methods according to the invention for calculating the transmission time of the packets from the time server 3 to the client station 1 will be described in the following. Despite the fact that the following methods are more elaborate concerning the determination of a time correction they do not require substantially more processing capabilities. Most steps do not require processing in real time and are performed at any available timeslot.

[0047] Referring to Figs. 7c and 7d a more accurate method for calculating the transmission time of the packets from the time server 3 to the client station 1 is shown. Fig. 7c illustrates the processing steps of the method. Fig. 7d shows time diagrams for the pulses sent from the time server I and received at the client station II. First, a round trip delay value for the first packet of the plurality of packets is determined. A first time difference 750 between the first packet and a following packet is determined for each of the following packets at the time the packets have been sent from the time server 3 (shown in Fig. 7d for the fourth packet). In the next step a second time difference 752 between the first packet and a following packet is determined for each of the following packets at the local time the packets have been received at the client station 1 (shown in Fig. 7d for the fourth packet).

[0048] Subtracting the second time difference 752 from the first time difference 750 for each packet results in a third time difference 754 for each packet. A positive value for the third time difference indicates that the corresponding packet traveled faster from the time server 3 to the client station 1 than the first packet. Accordingly, a negative value is indicative of a slower transmission. Therefore, a packet having a minimum travel time from the time server 3 to the client station 1 is indicated by having the maximum third time difference. The assumed travel time of the first packet being half of its round trip time is then corrected by subtracting the maximum third time difference from half of the round trip delay value of the first packet resulting in a minimum travel time of the corresponding packet. Data indicative of a correct time are then calculated by adding the minimum travel time to the predetermined time of the corresponding packet.

[0049] Accuracy of the above method is further improved by determining a minimum round trip delay value and a corresponding packet instead of the first packet for the assumption that the

travel time from the time server 3 to the client station 1 is half the round trip delay value, which is then corrected. This improved method is shown in Figs. 7e and 7f. It differs from the above method by including the step of determining a minimum round trip delay value and a corresponding packet. Then the first and second time differences are determined between the packet having the minimum round trip delay value and each of the other packets of the plurality of packets. Fig. 7f shows time diagrams for first time differences 760 and 770, with k being the packet having minimum round trip delay and $k - 2$ being a packet sent before the k^{th} packet and $k + 2$ being a packet sent after the k^{th} packet. Time differences of packets sent before the k^{th} packet are of negative value. Analogously, second time differences 762 and 772 and third time differences 764 and 774 are shown in Fig. 7f. As described above the packet having a minimum travel time from the time server 3 to the client station 1 is indicated by the maximum third time difference. A minimum travel time from the time server 3 to the client station 1 is then determined by subtracting the maximum third time difference from half of the round trip delay value of the packet having the minimum round trip delay. Data indicative of a correct time are then calculated by adding the minimum travel time to the predetermined time of the corresponding packet.

[0050] If it is known that a local time of a client station 1 is within known error bounds, preferably, the error being less than half the round trip delay value, the method according to the invention shown in Fig. 7g can be applied. A travel time for each of the plurality of packets from the time server to the client station is determined as the difference between the predetermined time the packet was sent from the server and the local time the packet was received at the client station. Preferably, a minimum travel time is determined and added to the predetermined time of the corresponding packet in order to obtain data indicative of a correct time. Optionally, two minimum travel times are determined based on the time for a packet to travel from the server to a client – based on the two different time sources – and a time for a packet to travel back to the server. The two minimum travel times are then used with the packet arrival times to determine data indicative of a correct time for synchronization of the client system.

[0051] The methods for synchronizing a timing device of a client station according to the invention are highly advantageous over the prior art. Transmitting a plurality of packets and

evaluating the transmission of the plurality of packets substantially increases accuracy if a sufficiently large number of packets are used having predetermined spacing therebetween. Furthermore, it allows statistical evaluation of the transmission through the network in order to detect any irregularities such as technical problems or tampering with the packets by an unauthorized third party.

[0052] Another advantage of the methods according to the invention is the reduction in real time processing required, only the steps of transmitting the plurality of packets and determining a time of receipt of the packets have to be performed in real time. These steps do not require much processing and can be performed parallel to other tasks.

[0053] The methods as disclosed above are easily implemented in the system shown in Fig. 1a, for example, as a software package for execution on processors 7 and 9 of the client station 1 and the time server 3, respectively. The software may be distributed on a CD-ROM or via the communications network 2. Alternatively, processors 11 and 13 for synchronizing the timing device 4 are connected to the client station 1 and the time server 3, respectively, as a peripheral device.

[0054] Of course, numerous other embodiments may be envisaged without departing from the spirit and scope of the claimed invention.